

Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Фізико-Технічний інститут

КРИПТОГРАФІЯ

КОМП’ЮТЕРНИЙ ПРАКТИКУМ №2

Виконала: Студентка 3-го курсу
Групи ФБ-93
Пономаренко Олександра Сергіївна

Київ 2021

Криптоаналіз шифру Віженера

Мета:

Засвоєння методів частотного криптоаналізу. Здобуття навичок роботи та аналізу поточкових шифрів гамування адитивного типу на прикладі шифру Віженера.

Завдання до виконання:

0. Уважно прочитати методичні вказівки до виконання комп'ютерного практикуму.

1. Самостійно підібрати текст для шифрування (2-3 кб) та ключі довжини $r = 2, 3, 4, 5$, а також довжини 10-20 знаків. Зашифрувати обраний відкритий текст шифром Віженера з цими ключами.
2. Підрахувати індекси відповідності для відкритого тексту та всіх одержаних шифртекстів і порівняти їх значення.
3. Використовуючи наведені теоретичні відомості, розшифрувати наданий шифртекст (згідно свого номеру варіанта).

Виконання роботи:

1) Для шифрування обрала текст, що знаходиться у SH.txt.

```
82      /*while (true) {
83          j2 = 0;
84          cout << "Enter the key word: ";
85          cin >> key;
86          key = RUS(key);
87          lengthOfKey = key.length();
88          for (int i = 0; i < lengthOfText; i++) {
89              j = alphabet.find(txt_before[i]);
90              OT[i] = j;
91          }
92          /*for (int i = 0; i < lengthOfText; i++) {
93              cout << OT[i] << "\t";
94          }*/
95          /*for (int i = 0; i < lengthOfText; i++, j2++) {
96              if (j2 == lengthOfKey) {
97                  j2 = -1;
98                  i--;
99              }
100             else {
101                 j1 = alphabet.find(key[j2]);
102                 key_mas[i] = j1;
103             }
104         }
105         /*cout << endl;
106         for (int i = 0; i < lengthOfText; i++) {
107             cout << key_mas[i] << "\t";
108         }*/
109         /*for (int i = 0; i < lengthOfText; i++) {
110             CT[i] = (OT[i] + key_mas[i]) % lengthOfAlphabet;
111         }
112         /*cout << endl;
113         for (int i = 0; i < lengthOfText; i++) {
114             cout << CT[i] << "\t";
115         }*/
116         /*for (int i = 0; i < lengthOfText; i++) {
117             txt_after[i] = alphabet[CT[i]];
118         }
119         cout << endl;
120         cout << "New .txt is ready. Enter the root to it: ";
121         cin >> file_name;
122         outputf(f1, txt_after, file_name);
123     }*/
```

Выбрать D:\Me\Кни\3 курс\5 семестр\крипта\n2\try1\Debug\try1.exe
Enter the root to .txt file: SH.txt
Enter the key word: ей
New .txt is ready. Enter the root to it: a1.txt
Enter the key word: как
New .txt is ready. Enter the root to it: a2.txt
Enter the key word: тело
New .txt is ready. Enter the root to it: a3.txt
Enter the key word: буква
New .txt is ready. Enter the root to it: a4.txt
Enter the key word: сашакашая
New .txt is ready. Enter the root to it: a5.txt
Enter the key word: теперьподул
New .txt is ready. Enter the root to it: a6.txt
Enter the key word: вконцебудет
New .txt is ready. Enter the root to it: a7.txt
Enter the key word: приветакадел
New .txt is ready. Enter the root to it: a8.txt
Enter the key word: абабабабаба
New .txt is ready. Enter the root to it: a9.txt
Enter the key word: pppрамдедушкау
New .txt is ready. Enter the root to it: a10.txt
Enter the key word: кукушкапелаиплас
New .txt is ready. Enter the root to it: a11.txt
Enter the key word: вернитедомойребен
New .txt is ready. Enter the root to it: a12.txt
Enter the key word: филинухалухухночью
New .txt is ready. Enter the root to it: a13.txt
Enter the key word: котсобакаиаквиумс
New .txt is ready. Enter the root to it: a14.txt
Enter the key word: курочкарябаснеслайц
New .txt is ready. Enter the root to it: a15.txt

Далі в циклі зашифрувала ВТ різними ключами(відповідно з довжинами 2-5, 10-20). Записала усі ШТ в a1-15.txt(зі збільшенням індексу - збільшена довжина ключа).

2) У наступному циклі зчитувала усі отримані ШТ та обраховувала індекси відповідності для кожного з них.

```
132     while (true) {
133         cout << "Enter the root to .txt file: ";
134         cin >> file_name;
135         txt_after = inputf(f, file_name);
136         lengthOfText = txt_after.length();
137         //cout << txt_after << endl;
138         f_quantityOfSym(quantityOfSym, lengthOfAlphabet, alphabet, txt_after);
139         index_OT = 0;
140         index_OT = f_index_OT(lengthOfText, quantityOfSym, lengthOfAlphabet, index_OT);
141         cout << "Index of Open Text: " << index_OT << endl;
142     }
```

CIPHER TEXT

Отримала такі значення:

```
D:\Me\Кні\3 курс\5 семестр\крипта\л2\try1\Debug\try1.exe
Enter the root to .txt file: SH.txt
Index of Open Text: 0.0570663
Enter the root to .txt file: a1.txt
Index of Open Text: 0.0461798 CIPHER TEXT
Enter the root to .txt file: a2.txt
Index of Open Text: 0.04287
Enter the root to .txt file: a3.txt
Index of Open Text: 0.038715
Enter the root to .txt file: a4.txt
Index of Open Text: 0.0363879
Enter the root to .txt file: a5.txt
Index of Open Text: 0.0356376
Enter the root to .txt file: a6.txt
Index of Open Text: 0.0348644
Enter the root to .txt file: a7.txt
Index of Open Text: 0.0342412
Enter the root to .txt file: a8.txt
Index of Open Text: 0.0347182
Enter the root to .txt file: a9.txt
Index of Open Text: 0.0457075
Enter the root to .txt file: a10.txt
Index of Open Text: 0.0331171
Enter the root to .txt file: a11.txt
Index of Open Text: 0.0338279
Enter the root to .txt file: a12.txt
Index of Open Text: 0.0344975
Enter the root to .txt file: a13.txt
Index of Open Text: 0.033131
Enter the root to .txt file: a14.txt
Index of Open Text: 0.0332255
Enter the root to .txt file: a15.txt
Index of Open Text: 0.0321062
```

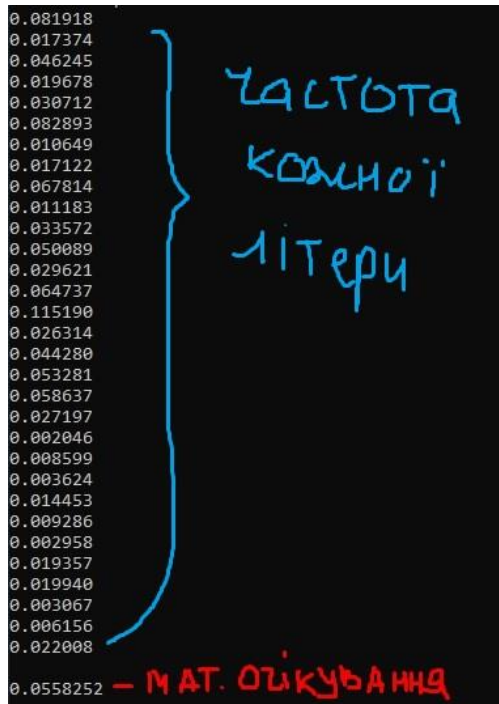
Дійсно, із збільшенням довжини ключа індекс зменшуються. Але в моєму випадку, у ШТ а9 стався стрибок. Я спочатку не зрозуміла, чому так, але потім подивившись на ключ для цього ШТ (а саме, абабабабабая), зрозуміла, що це через нього. (А) - має індекс 0, тому зміни під час шифрування не відбуваються.

Код для першої частини знаходиться в try1.cpp

3) Варіант 9:

ШТ знаходиться у СТ.txt

Для знаходження довжини ключа, нам потрібне теоретичне значення індексу відповідності (для знаходження використали отримані в першій лабораторній значення частот кожної літери):



Далі в циклі проводимо експеримент для різних довжин ключа, розбиваючи ШТ на блоки(довжина ключа=кількість блоків)

```
95 //знаходження довжини ключа
96 /*while (true) {
97     cout << "Enter the length of r: ";
98     cin >> lengthOfKey;
99     string* block = new string[lengthOfKey];
100     int extra;
101     //cout << txt_before << endl;
102     for (int rows = 0; rows < lengthOfKey; rows++) {
103         key = "";
104         extra = rows;
105         while (rows < lengthOfText) {
106             key = key + txt_before[rows];
107             rows = rows + lengthOfKey;
108         }
109         rows = extra;
110         block[rows] = key;
111     }
112     cout << endl;
113     int* quantityOfSym = new int[lengthOfAlphabet];
114     double index_OT = 0;
115     for (int i = 0; i < lengthOfKey; i++) {
116         //cout << block[i] << endl;
117         f_quantityOfSym(quantityOfSym, lengthOfAlphabet, alphabet, block[i]);
118         index_OT = f_index_OT(block[i].length(), quantityOfSym, lengthOfAlphabet, index_OT);
119         cout << "Index of Open Text: " << index_OT << endl;
120     }
121     delete[] block;
122 }*/
```


Продивляємося по всім значенням індексу, шукаючи найближче до математичного очікування. Саме при довжині 17, можемо бачити потрібні значення. Тому можна зробити висновок, що довжина нашого ключа дорівнює 17

```
D:\Me\Kni\3 курс\5 семестр\крипта\n2\try2.1\Debug\try2.1.exe
Enter the root to .txt file: CT.txt
Enter the length of r: 2      Enter the length of r: 11
Index of Open Text: 0.032795   Index of Open Text: 0.0331948
Index of Open Text: 0.0329832   Index of Open Text: 0.0327558
Enter the length of r: 3      Index of Open Text: 0.0330169
Index of Open Text: 0.0328095   Index of Open Text: 0.0328804
Index of Open Text: 0.0329217   Index of Open Text: 0.0328388
Index of Open Text: 0.0326769   Index of Open Text: 0.0338158
Enter the length of r: 4      Index of Open Text: 0.0327736
Index of Open Text: 0.0329149   Index of Open Text: 0.0315646
Index of Open Text: 0.0327417   Index of Open Text: 0.0327974
Index of Open Text: 0.0324196   Index of Open Text: 0.0323567
Index of Open Text: 0.0330301   Index of Open Text: 0.0324401
Enter the length of r: 5      Enter the length of r: 12
Index of Open Text: 0.0331977   Index of Open Text: 0.0329811
Index of Open Text: 0.0327644   Index of Open Text: 0.032617
Index of Open Text: 0.0332627   Index of Open Text: 0.033594
Index of Open Text: 0.0323741   Index of Open Text: 0.0322134
Index of Open Text: 0.0324539   Index of Open Text: 0.0327161
Enter the length of r: 6      Index of Open Text: 0.032086
Index of Open Text: 0.0327174   Index of Open Text: 0.0321851
Index of Open Text: 0.0324925   Index of Open Text: 0.0321214
Index of Open Text: 0.0325066   Index of Open Text: 0.031732
Index of Open Text: 0.0330247   Index of Open Text: 0.032886
Index of Open Text: 0.0331061   Index of Open Text: 0.032532
Index of Open Text: 0.0329858   Index of Open Text: 0.0337356
Enter the length of r: 7      Enter the length of r: 13
Index of Open Text: 0.0328611   Index of Open Text: 0.0330585
Index of Open Text: 0.0323098   Index of Open Text: 0.0323797
Index of Open Text: 0.0330681   Index of Open Text: 0.0323434
Index of Open Text: 0.0323965   Index of Open Text: 0.0322957
Index of Open Text: 0.0327215   Index of Open Text: 0.0312151
Index of Open Text: 0.0325746   Index of Open Text: 0.0336923
Index of Open Text: 0.0331644   Index of Open Text: 0.0331852
Enter the length of r: 8      Index of Open Text: 0.0320879
Index of Open Text: 0.0332057   Index of Open Text: 0.0340581
Index of Open Text: 0.0329211   Index of Open Text: 0.0325285
Index of Open Text: 0.0320658   Index of Open Text: 0.0338419
Index of Open Text: 0.0336035   Index of Open Text: 0.033684
Index of Open Text: 0.0322859   Index of Open Text: 0.0321295
Index of Open Text: 0.0327324   Enter the length of r: 14
Index of Open Text: 0.0328236   Index of Open Text: 0.0329571
Index of Open Text: 0.0330375   Index of Open Text: 0.0318779
Enter the length of r: 9      Index of Open Text: 0.0327262
Index of Open Text: 0.0331115   Index of Open Text: 0.0337576
Index of Open Text: 0.0330242   Index of Open Text: 0.0322732
Index of Open Text: 0.0320865   Index of Open Text: 0.0322539
Index of Open Text: 0.0335963   Index of Open Text: 0.0324274
Index of Open Text: 0.0323485   Index of Open Text: 0.0323213
Index of Open Text: 0.033221   Index of Open Text: 0.0323213
Index of Open Text: 0.0319222   Index of Open Text: 0.0328997
Index of Open Text: 0.0322369   Index of Open Text: 0.0332757
Index of Open Text: 0.0327429   Index of Open Text: 0.0318008
Enter the length of r: 10      Index of Open Text: 0.0338347
Index of Open Text: 0.0327755   Index of Open Text: 0.0328901
Index of Open Text: 0.0322262   Index of Open Text: 0.0336323
Index of Open Text: 0.0333005   Enter the length of r: 15
Index of Open Text: 0.0324371   Index of Open Text: 0.032179
Index of Open Text: 0.0325254   Index of Open Text: 0.0341344
Index of Open Text: 0.0338578   Index of Open Text: 0.0320988
Index of Open Text: 0.033223   Index of Open Text: 0.0323117
Index of Open Text: 0.0329277   Index of Open Text: 0.0317483
Index of Open Text: 0.0322043   Index of Open Text: 0.0326541
Index of Open Text: 0.0330557   Index of Open Text: 0.0308756
Index of Open Text: 0.0327755   Index of Open Text: 0.0340018
Index of Open Text: 0.0322262   Index of Open Text: 0.0322012
Index of Open Text: 0.0333005   Index of Open Text: 0.0336373
Index of Open Text: 0.0324371   Index of Open Text: 0.0334488
Index of Open Text: 0.0325254   Index of Open Text: 0.0325862
Index of Open Text: 0.0338578   Index of Open Text: 0.0338626
Index of Open Text: 0.033223   Index of Open Text: 0.0311656
Index of Open Text: 0.0329277   Index of Open Text: 0.0325196
Index of Open Text: 0.0322043
Index of Open Text: 0.0330557

Enter the length of r: 17
Index of Open Text: 0.0490922
Index of Open Text: 0.055206
Index of Open Text: 0.0523692
Index of Open Text: 0.051277
Index of Open Text: 0.0546103
Index of Open Text: 0.0592061
Index of Open Text: 0.0529366
Index of Open Text: 0.0529791
Index of Open Text: 0.0600004
Index of Open Text: 0.05556
Index of Open Text: 0.0545031
Index of Open Text: 0.0608346
Index of Open Text: 0.0533765
Index of Open Text: 0.0621608
Index of Open Text: 0.0538614
Index of Open Text: 0.0520931
Index of Open Text: 0.0614763
Enter the length of r: 18
Index of Open Text: 0.0328798
Index of Open Text: 0.0323867
Index of Open Text: 0.0324822
Index of Open Text: 0.0340805
Index of Open Text: 0.0326731
Index of Open Text: 0.0318137
Index of Open Text: 0.0319729
Index of Open Text: 0.0313045
Index of Open Text: 0.0320365
Index of Open Text: 0.0323389
Index of Open Text: 0.0333097
Index of Open Text: 0.033071
Index of Open Text: 0.033453
Index of Open Text: 0.0326022
Index of Open Text: 0.0335785
Index of Open Text: 0.0315939
Index of Open Text: 0.0315939
Index of Open Text: 0.0341386
Enter the length of r: 19
Index of Open Text: 0.0333121
Index of Open Text: 0.0332448
Index of Open Text: 0.0314147
Index of Open Text: 0.0334758
Index of Open Text: 0.0315568
Index of Open Text: 0.0327117
Index of Open Text: 0.032463
Index of Open Text: 0.0321431
Index of Open Text: 0.0345419
Index of Open Text: 0.0328184
Index of Open Text: 0.0326051
Index of Open Text: 0.0338489
Index of Open Text: 0.0332093
Index of Open Text: 0.0334047
Index of Open Text: 0.0323919
Index of Open Text: 0.0331559
Index of Open Text: 0.0322675
Index of Open Text: 0.0316634
Index of Open Text: 0.0345774
Enter the length of r: 20
Index of Open Text: 0.032406
Index of Open Text: 0.0325434
Index of Open Text: 0.0328961
Index of Open Text: 0.0317401
Index of Open Text: 0.0322887
Index of Open Text: 0.0331818
Index of Open Text: 0.0345619
Index of Open Text: 0.033241
Index of Open Text: 0.0326101
Index of Open Text: 0.0332016
Index of Open Text: 0.0327087
Index of Open Text: 0.0317229
Index of Open Text: 0.0326101
Index of Open Text: 0.0316046
Index of Open Text: 0.0320975
Index of Open Text: 0.0344634
Index of Open Text: 0.0323341
Index of Open Text: 0.0319003
Index of Open Text: 0.0309342
Index of Open Text: 0.0321369
```


Enter the length of r: 21

Index of Open Text: 0.0353753
Index of Open Text: 0.0322437
Index of Open Text: 0.0332642
Index of Open Text: 0.0337853
Index of Open Text: 0.0316358
Index of Open Text: 0.0316575
Index of Open Text: 0.0307672
Index of Open Text: 0.0307889
Index of Open Text: 0.0324825
Index of Open Text: 0.0323306
Index of Open Text: 0.0308975
Index of Open Text: 0.0334162
Index of Open Text: 0.0343281
Index of Open Text: 0.0352835
Index of Open Text: 0.0325043
Index of Open Text: 0.0313752
Index of Open Text: 0.0348492
Index of Open Text: 0.0318132
Index of Open Text: 0.0326562
Index of Open Text: 0.0321786
Index of Open Text: 0.0354572
Enter the length of r: 22

Index of Open Text: 0.0320654
Index of Open Text: 0.0343172
Index of Open Text: 0.032445
Index of Open Text: 0.0303594
Index of Open Text: 0.0336773
Index of Open Text: 0.0341729
Index of Open Text: 0.033457
Index of Open Text: 0.031715
Index of Open Text: 0.0324786
Index of Open Text: 0.0306888
Index of Open Text: 0.0321922
Index of Open Text: 0.0350081
Index of Open Text: 0.0325502
Index of Open Text: 0.0341729
Index of Open Text: 0.0362491
Index of Open Text: 0.0326457
Index of Open Text: 0.0341252
Index of Open Text: 0.0309275
Index of Open Text: 0.0300445
Index of Open Text: 0.0337911
Index of Open Text: 0.0337434
Index of Open Text: 0.0323116
Enter the length of r: 23

Index of Open Text: 0.031504
Index of Open Text: 0.0325174
Index of Open Text: 0.0328031
Index of Open Text: 0.0309071
Index of Open Text: 0.0322317
Index of Open Text: 0.032881
Index of Open Text: 0.0343354
Index of Open Text: 0.0322446
Index of Open Text: 0.0316863
Index of Open Text: 0.0314785
Index of Open Text: 0.0358158
Index of Open Text: 0.0313746
Index of Open Text: 0.0343873
Index of Open Text: 0.0335563
Index of Open Text: 0.0334332
Index of Open Text: 0.0327007
Index of Open Text: 0.0334855
Index of Open Text: 0.0341918
Index of Open Text: 0.0322298
Index of Open Text: 0.0336686
Index of Open Text: 0.0319421
Index of Open Text: 0.0313403
Index of Open Text: 0.0323083

Enter the length of r: 24

Index of Open Text: 0.0314269
Index of Open Text: 0.0324022
Index of Open Text: 0.0327143
Index of Open Text: 0.0324306
Index of Open Text: 0.0313524
Index of Open Text: 0.0320901
Index of Open Text: 0.0342464
Index of Open Text: 0.0326009
Index of Open Text: 0.0313241
Index of Open Text: 0.033679
Index of Open Text: 0.0326292
Index of Open Text: 0.0344734
Index of Open Text: 0.0340479
Index of Open Text: 0.0321185
Index of Open Text: 0.0340195
Index of Open Text: 0.031239
Index of Open Text: 0.0338208
Index of Open Text: 0.0325441
Index of Open Text: 0.0312106
Index of Open Text: 0.0332818
Index of Open Text: 0.0310971
Index of Open Text: 0.0326576
Index of Open Text: 0.0321469
Index of Open Text: 0.0337641
Enter the length of r: 25

Index of Open Text: 0.0344363
Index of Open Text: 0.0311586
Index of Open Text: 0.0322615
Index of Open Text: 0.0313424
Index of Open Text: 0.0306684
Index of Open Text: 0.0343142
Index of Open Text: 0.032139
Index of Open Text: 0.0334564
Index of Open Text: 0.0310361
Index of Open Text: 0.0316181
Index of Open Text: 0.0316509
Index of Open Text: 0.0352328
Index of Open Text: 0.0326391
Index of Open Text: 0.0336889
Index of Open Text: 0.0316819
Index of Open Text: 0.0358813
Index of Open Text: 0.033689
Index of Open Text: 0.0325156
Index of Open Text: 0.0333493
Index of Open Text: 0.0329479
Index of Open Text: 0.0332258
Index of Open Text: 0.0313422
Index of Open Text: 0.0329787
Index of Open Text: 0.0314657
Index of Open Text: 0.0332257
Enter the length of r: 26

Index of Open Text: 0.0333831
Index of Open Text: 0.0326868
Index of Open Text: 0.0325541
Index of Open Text: 0.0335828
Index of Open Text: 0.0330518
Index of Open Text: 0.0332509
Index of Open Text: 0.0329523
Index of Open Text: 0.0313262
Index of Open Text: 0.0331513
Index of Open Text: 0.03272
Index of Open Text: 0.0336823
Index of Open Text: 0.031094
Index of Open Text: 0.0315253
Index of Open Text: 0.0327863
Index of Open Text: 0.0330518
Index of Open Text: 0.0359992
Index of Open Text: 0.0329882
Index of Open Text: 0.0306128
Index of Open Text: 0.0335234
Index of Open Text: 0.0331555
Index of Open Text: 0.0330551
Index of Open Text: 0.0345941
Index of Open Text: 0.0327206
Index of Open Text: 0.0325198
Index of Open Text: 0.0364676
Index of Open Text: 0.0323191

Enter the length of r: 27

Index of Open Text: 0.0346492
Index of Open Text: 0.0312529
Index of Open Text: 0.0308952
Index of Open Text: 0.0329692
Index of Open Text: 0.0306092
Index of Open Text: 0.033434
Index of Open Text: 0.034471
Index of Open Text: 0.0304662
Index of Open Text: 0.0324685
Index of Open Text: 0.0325759
Index of Open Text: 0.0334698
Index of Open Text: 0.0338631
Index of Open Text: 0.0329335
Index of Open Text: 0.0318073
Index of Open Text: 0.0326728
Index of Open Text: 0.0293551
Index of Open Text: 0.0312302
Index of Open Text: 0.0326006
Index of Open Text: 0.0330695
Index of Open Text: 0.0332858
Index of Open Text: 0.0329974
Index of Open Text: 0.0338628
Index of Open Text: 0.0342956
Index of Open Text: 0.0316991
Index of Open Text: 0.031086
Index of Open Text: 0.0323121
Index of Open Text: 0.0323843
Enter the length of r: 28

Index of Open Text: 0.031104
Index of Open Text: 0.0318423
Index of Open Text: 0.0309922
Index of Open Text: 0.034586
Index of Open Text: 0.0313787
Index of Open Text: 0.0347019
Index of Open Text: 0.0309922
Index of Open Text: 0.0325379
Index of Open Text: 0.0339677
Index of Open Text: 0.0309149
Index of Open Text: 0.0306444
Index of Open Text: 0.0365567
Index of Open Text: 0.0331563
Index of Open Text: 0.0330403
Index of Open Text: 0.0348951
Index of Open Text: 0.0306445
Index of Open Text: 0.0342768
Index of Open Text: 0.0322288
Index of Open Text: 0.0310695
Index of Open Text: 0.030799
Index of Open Text: 0.0339677
Index of Open Text: 0.0327608
Index of Open Text: 0.0300763
Index of Open Text: 0.0339677
Index of Open Text: 0.0328471
Index of Open Text: 0.0319969
Index of Open Text: 0.031881
Index of Open Text: 0.0331948

Enter the length of r: 29

Index of Open Text: 0.033731
Index of Open Text: 0.0327856
Index of Open Text: 0.0328678
Index of Open Text: 0.0336905
Index of Open Text: 0.02896
Index of Open Text: 0.0361152
Index of Open Text: 0.0336247
Index of Open Text: 0.0318396
Index of Open Text: 0.0348699
Index of Open Text: 0.0337492
Index of Open Text: 0.0327944
Index of Open Text: 0.0301792
Index of Open Text: 0.0303452
Index of Open Text: 0.0323792
Index of Open Text: 0.035119
Index of Open Text: 0.0327944
Index of Open Text: 0.0354096
Index of Open Text: 0.0339982
Index of Open Text: 0.0327529
Index of Open Text: 0.0327113
Index of Open Text: 0.0309264
Index of Open Text: 0.0337491
Index of Open Text: 0.033251
Index of Open Text: 0.0330019
Index of Open Text: 0.0305943
Index of Open Text: 0.0321717
Index of Open Text: 0.0323377
Index of Open Text: 0.035202
Index of Open Text: 0.0334586
Enter the length of r: 30

Index of Open Text: 0.0331296
Index of Open Text: 0.0319345
Index of Open Text: 0.0314915
Index of Open Text: 0.03096
Index of Open Text: 0.0308272
Index of Open Text: 0.033396
Index of Open Text: 0.0310044
Index of Open Text: 0.0332189
Index of Open Text: 0.0315359
Index of Open Text: 0.0353448
Index of Open Text: 0.0336176
Index of Open Text: 0.0330418
Index of Open Text: 0.0341933
Index of Open Text: 0.0306058
Index of Open Text: 0.0334403
Index of Open Text: 0.031713
Index of Open Text: 0.0364964
Index of Open Text: 0.0312702
Index of Open Text: 0.0328645
Index of Open Text: 0.0322002
Index of Open Text: 0.0300272
Index of Open Text: 0.0304285
Index of Open Text: 0.0337061
Index of Open Text: 0.0323774
Index of Open Text: 0.0332632
Index of Open Text: 0.0336679
Index of Open Text: 0.0318795
Index of Open Text: 0.0323713
Index of Open Text: 0.0319689
Index of Open Text: 0.0321477

Знаходимо у кожному блоці літеру, що найчастіше зустрічається:

```
Консоль отладки Microsoft Visual Studio
Enter the root to .txt file: CT.txt
Enter the length of r: 17

свхзфпфйвпвфпккрдйужтолпкэтгнлмпрнпфтрмзш
впусгкпвуурптнеезпкрдунрзщещдээзюпчсрулрс
Key[a]: пьоноьогелытжэцлк
Key[b]: оынмнщнвдксьеьхкй
Key[v]: ньмлмшмбгйщрдыфйи
Key[r]: мщклчлавишпгъуиз
Key[d]: лшкйкцябзчовщтэж
Key[e]: кчиййхийюажцнбшсже
Key[j]: йцизифизеяхмачред
Key[z]: ихэжзузьюдфляцпдг
Key[i]: зфжежтжыэгукюхогв
Key[y]: жуедесеььвтйэфнвб
Key[k]: етдгдрдшыбсиьумба
Key[l]: дсгвгпгшьарзытлая
Key[m]: грвбвовчщяпжъскяю
Key[n]: впбабнбцшюоещрийюэ
Key[o]: боаяамахчэндшпиэь
Key[p]: аняюяляфцьмгчозьы
Key[r]: ямяююкюухылвцнжъь
Key[s]: юлэьэйтфькбхмьещ
Key[t]: экъьыьсущйафлдщш
Key[u]: ьйьпызыртшияукгшч
Key[f]: ьийъъжъпсчзютйвчц
Key[x]: ьзщщещорцжэсибцх
Key[c]: щжшчшдшнпхерьзахф
Key[ch]: шечцгчмофдыпжяфу
Key[sh]: чдцццвцлнугъоеют
Key[sh]: цгхфхбхкмтвщндэтс
Key[ъ]: хвфуфафйлсбшмгьср
Key[y]: фбутуяуйкращлвырп
Key[ь]: уатстютэйпяцкбьпо
Key[э]: тьсрсэсжиоухйащон
Key[ю]: сюрпрьрезнэфияшнм
Key[я]: рэпопыпджмьюзючмл

Max: 34
Index of Open Text: 0.0490922
```

І далі просто за схемою шифру Цезаря знаходимо можливі ключі. В даному варіанті просто обрати за найчастішу літеру (О) було недостатньо. Тому треба проводити частотний аналіз.

Дивлячись на ключ key[o], я помітила слово эндшпиэь. Було не складно догадатися, що насправді там эндшпиль. Але далі вже пішли складнощі. Розшифрувавши ШТ за допомогою key[o], отримала незв'язний текст, де на кожному 10-ому місці (ключ довжини 17) було зрозуміле слово, або навіть декілька, отже частина ключа эндшпиль була вгадана правильно. Далі дивлячись на неповноцінний ВТ намагалася закінчити або виправити слова, що вже були зрозумілими. Таким чином я і дійшла до значення ключа: войнамагаэндшпиль (key.txt).

Отриманий ВТ знаходиться в OT.txt. Код для другої частини лабораторної знаходиться в try2.1.cpp

Висновок:

За цю лабораторну роботу ми дізналися більше про поліалфавітні підстановки, а саме про шифр Віженера. Навчилися методу розшифрування для відповідних ШТ та практично продивилися залежність значення індексу відповідності від довжини ключа шифрування.